Springboard DSC Program

Capstone Milestone report

Credit Card Fraud Detection

Anonymized credit card transactions labeled as fraudulent or genuine

By Harinee Madhusudhan

Febrauary2020

The problem chosen for this project is to predict fraudulent credit card transactions by using machine learning models. A dataset containing thousands of individual transactions and their respective labels was obtained from Kaggle website.

The objective was to create simple and commonly used machine learning models like logistic regression, KNN, random forest and maybe others to compare how they perform regarding for the task of predicting fraudulent credit card transactions.

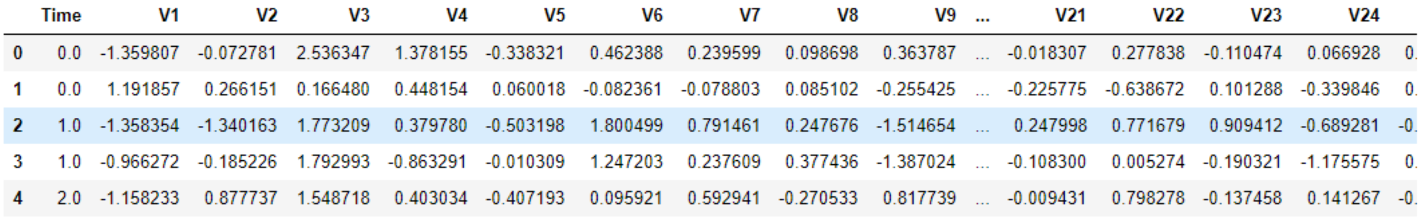
Description of the dataset, how obtained, cleaned, and wrangled it

Located some normalized data based on prior analysis. The datasets available from https://www.kaggle.com/mlg-ulb/creditcardfraud, contain transactions made by credit cards in September 2013 by European cardholders. This dataset present transactions that occurred in two days, where we have 492 frauds out of a total of 284,807 transactions. The datasets are highly unbalanced, the positive class (frauds) account for 0.172% of all transactions.

It contains only numerical input variables which are the result of a PCA transformation. Features V1, V2, ... V28 are the principal components obtained with PCA, the only features which have not been transformed with PCA are 'Time' and 'Amount'. Feature 'Time' contains the seconds elapsed between each transaction and the first transaction in the dataset. The feature 'Amount' is the transaction Amount. Feature 'Class' represents the class labelling, it takes value 1 in case of fraud and 0 otherwise.

The objective was to create simple and commonly used machine learning models like logistic regression, KNN, random forest and maybe others to compare how they perform regarding for the task of predicting fraudulent credit card transaction

Usually accuracy is given priority when anyone is assessing a model performance. However the data in this case is highly unbalanced, so the aim is to come closer to the classifier and plot the data whether under the curve or not The aim is to focus on the Area Under the two distributions.

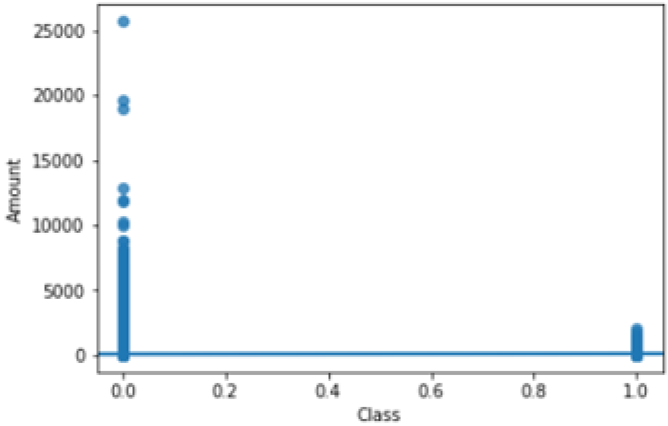


	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V24	
0	0.0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	...	-0.018307	0.277838	-0.110474	0.066928	0.
1	0.0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	...	-0.225775	-0.638672	0.101288	-0.339846	0.
2	1.0	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676	-1.514654	...	0.247998	0.771679	0.909412	-0.689281	-0.
3	1.0	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436	-1.387024	...	-0.108300	0.005274	-0.190321	-1.175575	0.
4	2.0	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.270533	0.817739	...	-0.009431	0.798278	-0.137458	0.141267	-0.

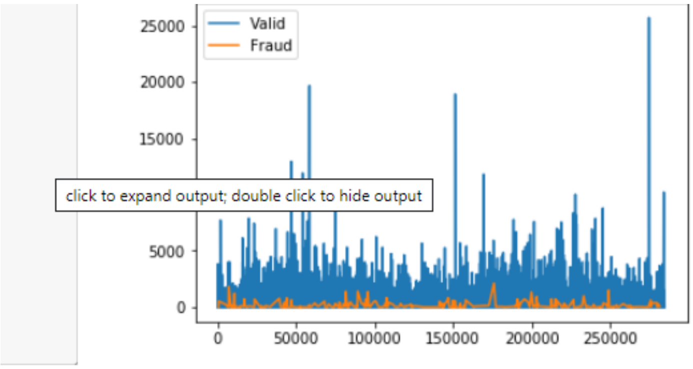
Exploratory Visualization

The interquartile range method found 31904 outliers, which represents 11.2% of the observations. Removing them from the dataset would be a bad idea due to the loss of a large amount of information for the machine learning models.

Tried to find what is the transaction amounts in different transaction classes like fraudulent and non-fraudulent Looks like fraudulent transactions are of lower denominations only.



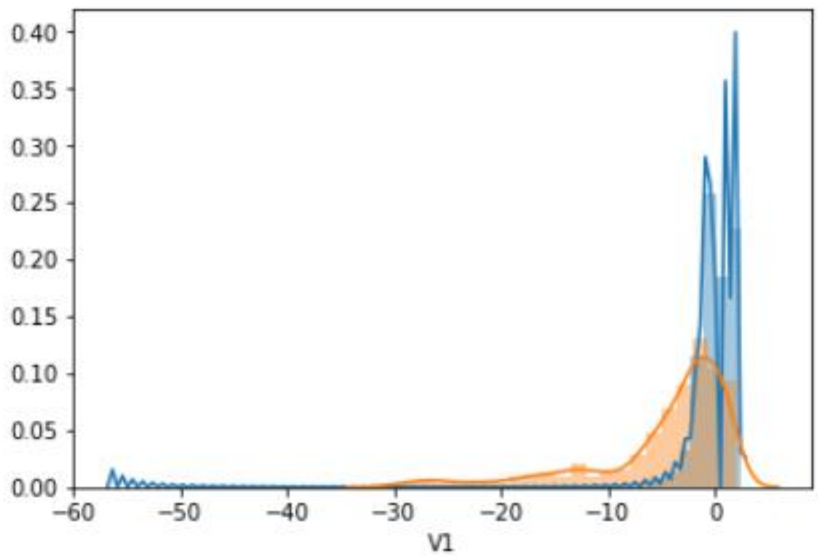
Is there a specific time period when fraudulent transactions occur?



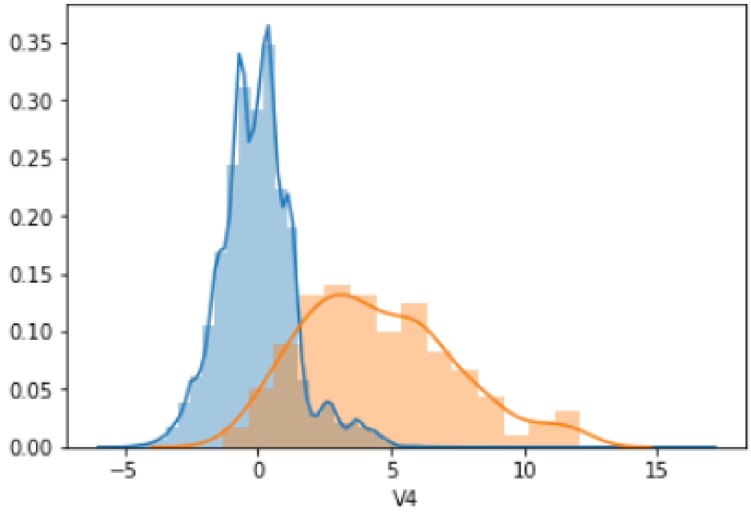
From the data available, it looks like the fraudulent transactions occur continuously, interspersed between the good transactions

Which of the 28 variables have more impact in the determination of fraudulent transactions?

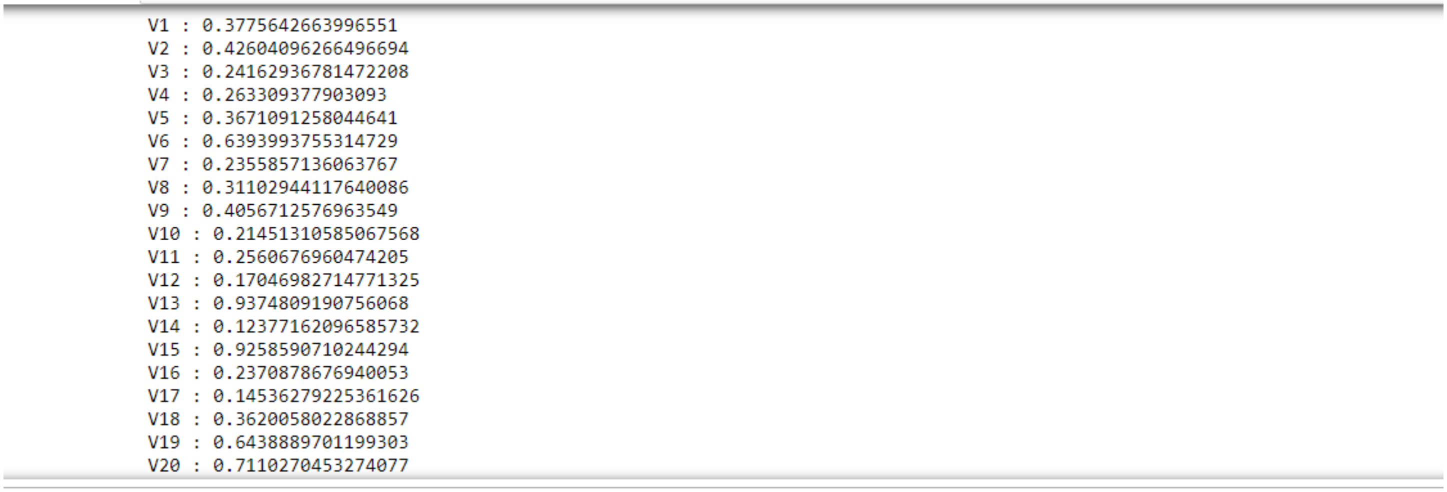
To respond to this question, we draw the histogram distributions for good and fraudulent transactions with a specific variable and see if the distributions overlap or are distinct. The more distinct they are, the more effect the variable has on the determination of fraudulent transactions.



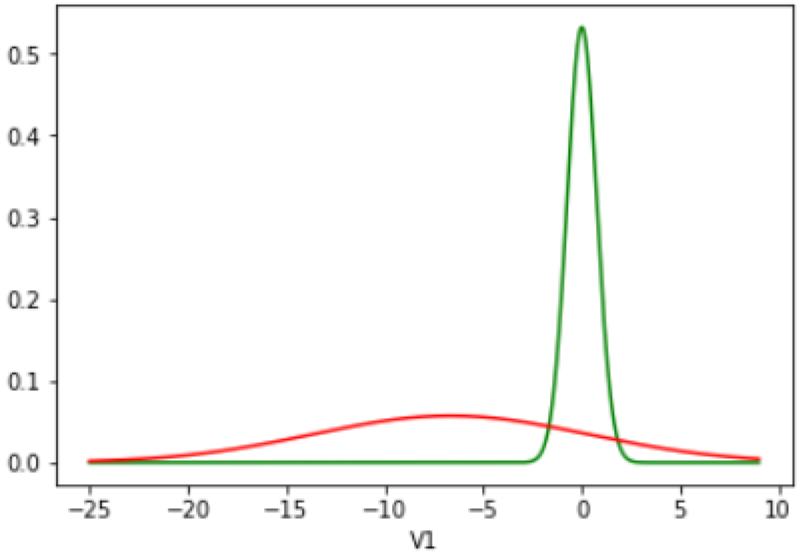
Also aimed to find a process to compute the overlap area between the two distributions so that which variables are particularly significant in terms of explaining the answer to the project question.

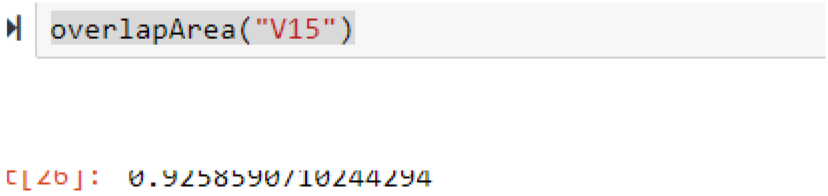


Below is the values of the overlap areas of each variables



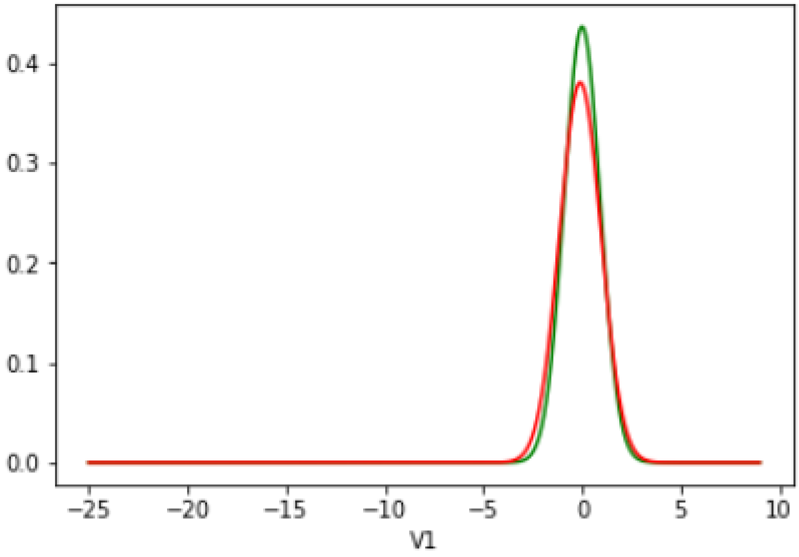
```
V1 : 0.3775642663996551
V2 : 0.42604096266496694
V3 : 0.24162936781472208
V4 : 0.263309377903093
V5 : 0.3671091258044641
V6 : 0.6393993755314729
V7 : 0.2355857136063767
V8 : 0.31102944117640086
V9 : 0.4056712576963549
V10 : 0.21451310585067568
V11 : 0.2560676960474205
V12 : 0.17046982714771325
V13 : 0.9374809190756068
V14 : 0.12377162096585732
V15 : 0.9258590710244294
V16 : 0.2370878676940053
V17 : 0.14536279225361626
V18 : 0.3620058022868857
V19 : 0.6438889701199303
V20 : 0.7110270453274077
```





```
overlapArea("V15")
```

```
t[26]:  0.9258590710244294
```



Working on this project helped to understand what the ability would be to work with data at an abstracted level and trying out different algorithms. This can be a initialization of a larger topic to handle the credit card fraud detection globally. To maximize the precision level more data could be collected which would in turn help the complexity and usage of advanced machine learning techniques.

Future proposal:

Using different types of classifier to understand the train and test data and then compute the confusion matrix and ROC curve. Each classifier should be appended to a list inside a tuple with its name as the first element and the classifier definition as the second element. This list will serve as an input for a function that plots the ROC curve, calculates the AUC and shows the confusion matrix for each classifier. (Logistic Regression, KNN KNeighbors Classifier, Decision Tree classifier, and Random Forest Classifier.)